| REPORT DOCUMENTATION PAGE | *Form Approved*<br>*OMB No. 0704-0188* |
|---|---|

| | | |
|---|---|---|
| 1. REPORT DATE *(DD-MM-YYYY)*<br>15-08-2007 | 2. REPORT TYPE<br>Final | 3. DATES COVERED<br>June 2004-Sept.2007 |

| | |
|---|---|
| 4. TITLE AND SUBTITLE<br>Collaborative Human-Computer Decision Making for Command and Control Resource Allocation | 5a. CONTRACT NUMBER<br>N000140410543 |
| | 5b. GRANT NUMBER<br>BAA04-001 |
| | 5c. PROGRAM ELEMENT NUMBER<br>N/A |
| 6. AUTHOR(S)<br>Cummings, Mary L. | 5d. PROJECT NUMBER<br>N/A |
| | 5e. TASK NUMBER<br>N/A |
| | 5f. WORK UNIT NUMBER<br>N/A |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>MIT<br>77 Massachusetts Ave., 33-305<br>Cambridge, MA 02139 | 8. PERFORMING ORGANIZATION REPORT NUMBER<br>6896251 |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)<br>Office of Naval Research | 10. SPONSOR/MONITOR'S ACRONYM(S)<br>ONR |
| | 11. SPONSOR/MONITOR'S REPORT NUMBER(S)<br>N000140410543 |

12. DISTRIBUTION/AILABILITY STATEMENT

Approved for Public Release, distribution unlimited

13. SUPPLEMENTARY NOTES

In command and control domains, mission goals are driven by human intentions and actions, and then executed and communicated through advanced, automated technology. Automation can make computations quickly and accurately based on a predetermined set of rules and conditions, which is especially effective for planning and making decisions in large problem spaces like those in command and control domains. However, computer optimization algorithms can only take into account those quantifiable variables identified in the design stages that were deemed to be critical. The focus of this research program is the development of a collaborative human-computer decision-making model that demonstrates not only what decision making functions should always be assigned to humans or computers, but what functions can best be served in a mutually supportive human-computer decision making environment. It is possible that when the human and computer collaborate, they can discover solutions superior to the one either would have determined independently of the other. This research effort investigates the strengths and limitations of both humans and computers in command and control resource allocation problems, and examines how humans and computer can work together collaboratively to promote efficient, effective, and robust decision making.

15. SUBJECT TERMS

Supervisory Control; levels of automation, decision support system; human-computer interaction, mission planning

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Mary L. Cummings |
| U | U | U | UP | 19 | 19b. TELEPHONE NUMBER *(Include area code)*<br>617-252-1512 |

# Collaborative Human-Computer Decision Making for Command and Control Resource Allocation

## Final Report

Mary Cummings
MIT
77 Massachusetts Ave 33-305
Cambridge, MA 02139
Phone: 617-252-1512 Fax: Email: **missyc@mit.edu**
Website: **http://halab.mit.edu**

Award Number: N000140410543

### 1.0 Summary

In command and control domains, mission goals are driven by human intentions and actions, and then executed and communicated through advanced, automated technology. Because the use of complex, automated systems will only increase in the future, more research needs to specifically address how humans and automation can collaborate with automation in mission planning and decision making in dynamic and uncertain environments. Automation can make computations quickly and accurately based on a predetermined set of rules and conditions, which is especially effective for planning and making decisions in large problem spaces like those in command and control domains. However, computer optimization algorithms can only take into account those quantifiable variables identified in the design stages that were deemed to be critical. In contrast, humans can reason inductively and generate conceptual representations based on both abstract and factual information, thus integrating qualitative and quantitative information. While humans are not able to integrate information as quickly as a computer and are sometimes susceptible to flawed decision making due to biased heuristics such as anchoring and recency (Tversky & Kahneman, 1974), their ability to leverage inductive reasoning and effective heuristics such as bounded rationality (Simon et al., 1986) and fast frugal decision making (Gigerenzer & Todd, 1999) can compensate for optimization algorithms' inherent limitations.

The focus of this research program was the development of a collaborative human-computer decision-making model that demonstrates not only what decision making functions should always be assigned to humans or computers, but what functions can best be served in a mutually supportive human-computer decision making environment. It is possible that when the human and computer collaborate, they can discover solutions superior to the one either would have determined independently of the other. This research effort investigated the strengths and limitations of both humans and computers in command and control resource allocation problems, and examines how humans and computer can work together collaboratively to promote efficient, effective, and robust decision making.

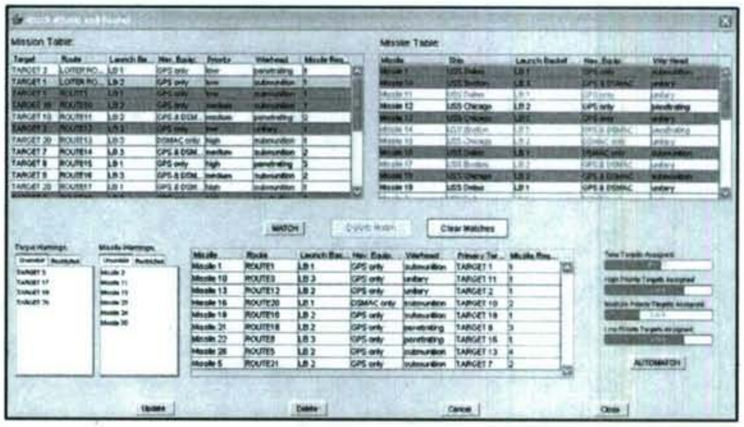
## 2.0 Research Accomplishments

1. Developed a Tactical Tomahawk mission planning simulation test bed, including the development of a heuristic-search algorithm to match preplanned missions to various missile loadouts.
2. Developed increasingly automated decision support tools for the Tomahawk mission planner.
3. Conducted an experiment with actual Naval personnel to examine human-computer collaboration performance and related cognitive strategies for Tomahawk mission planning.
4. Developed a preliminary model to capture cognitive strategies post hoc and determine how automation does or does not support effective strategies. This tool is caused Tracking Resource Allocation Cognitive Strategies (TRACS).
   - A technology disclosure was submitted to the MIT Intellectual Property office.
   - This tool has been used in three experiments, two ONR and one NASA.
5. Developed an initial prototype for a real-time Tomahawk/UAV retargeting decision support tool within a larger simulation environment.
6. Developed the Human-Automation Collaboration Taxonomy (HACT) to allow for better descriptive models of human-automation interaction.
7. Acquired the Mobile Advanced Command and Control Station, a mobile experimental test bed, through an ONR DURIP
8. Published one journal article, one book chapter, and 5 conference papers.

## 3.0 Completed Experiments & Performance Data

### 3.1 Experiment #1[1]

A pilot experiment was conducted in October 2005 to determine how operators would search a complex mission planning solution space using three different interfaces, which represent increasing levels of automation ranging from mostly operator-directed to mostly automation-directed. The focus of the research was to determine how solutions would be generated, and the effectiveness of the combined performance of the human and the computer for the overall mission goal. The first matching interface (Figure 1) allows for manual matching and computer generated matching in the mission-missile resource allocation. The operator selects a mission in the mission table and a missile in the missile table (among those which have been



Mission Table:

| Target | Route | Launch Ba... | Nav. Equip. | Priority | Warhead | Missile Req... |
|---|---|---|---|---|---|---|
| TARGET 2 | LOITER RO... | LB 1 | GPS only | low | penetrating | 1 |
| TARGET 1 | LOITER RO... | LB 2 | GPS only | low | submunition | 1 |
| TARGET 5 | ROUTE5 | LB 1 | GPS only | low | submunition | 1 |
| TARGET 18 | ROUTE10 | LB 2 | GPS only | medium | submunition | 1 |
| TARGET 10 | ROUTE11 | LB 2 | GPS & DSM... | medium | penetrating | 2 |
| TARGET 3 | ROUTE12 | LB 3 | GPS only | low | unitary | 1 |
| TARGET 20 | ROUTE13 | LB 3 | DSMAC only | high | submunition | 1 |
| TARGET 7 | ROUTE14 | LB 3 | GPS & DSM... | medium | submunition | 2 |
| TARGET 8 | ROUTE15 | LB 1 | GPS only | high | penetrating | 3 |
| TARGET 9 | ROUTE16 | LB 3 | GPS & DSM... | medium | submunition | 2 |
| TARGET 20 | ROUTE17 | LB 1 | GPS & DSM... | high | submunition | 1 |

Missile Table:

| Missile | Ship | Launch Basket | Nav. Equip. | War Head |
|---|---|---|---|---|
| Missile 1 | USS Dallas | LB 1 | GPS only | submunition |
| Missile 10 | USS Boston | LB 3 | GPS & DSMAC | unitary |
| Missile 11 | USS Dallas | LB 1 | GPS only | unitary |
| Missile 12 | USS Chicago | LB 2 | GPS only | penetrating |
| Missile 13 | USS Chicago | LB 2 | GPS only | unitary |
| Missile 14 | USS Boston | LB 1 | GPS & DSMAC | penetrating |
| Missile 15 | USS Chicago | LB 2 | DSMAC only | unitary |
| Missile 16 | USS Dallas | LB 1 | DSMAC only | submunition |
| Missile 17 | USS Boston | LB 3 | GPS & DSMAC | unitary |
| Missile 18 | USS Chicago | LB 2 | GPS & DSMAC | submunition |
| Missile 19 | USS Dallas | LB 1 | GPS & DSMAC | unitary |

Target Warnings:

TARGET 5
TARGET 17
TARGET 19
TARGET 2[illegible]

Missile Warnings:

Missile 2
Missile 11
Missile 15
Missile 2[illegible]
Missile 24
Missile 30

| Missile | Route | Launch Bas... | Nav. Equip. | Warhead | Primary Tar... | Missile Req... |
|---|---|---|---|---|---|---|
| Missile 1 | ROUTE1 | LB 1 | GPS only | submunition | TARGET 1 | 1 |
| Missile 10 | ROUTE3 | LB 3 | GPS only | unitary | TARGET 11 | 1 |
| Missile 13 | ROUTE12 | LB 2 | GPS only | unitary | TARGET 2 | 1 |
| Missile 16 | ROUTE20 | LB 1 | DSMAC only | submunition | TARGET 10 | 2 |
| Missile 18 | ROUTE10 | LB 2 | GPS only | submunition | TARGET 18 | 1 |
| Missile 21 | ROUTE18 | LB 2 | GPS only | penetrating | TARGET 8 | 3 |
| Missile 22 | ROUTE8 | LB 3 | GPS only | penetrating | TARGET 16 | 1 |
| Missile 26 | ROUTE5 | LB 2 | GPS only | submunition | TARGET 13 | 4 |
| Missile 5 | ROUTE21 | LB 2 | GPS only | submunition | TARGET 7 | 2 |

**Figure 1 - StrikeView Matching Interface 1.**

[1] Described more in detail in Bruni, S., Cummings M.L., "Human Interaction with Mission Planning Search Algorithms". ASNE Human Systems Integration Conference. 2005.

filtered out by the computer as satisfying hard constraints). The tables display the primary characteristics of the missions (Target, Route, Launch Basket, Navigation Equipment Required, Priority, Warhead Required, and Number of Missiles Required), and those of the missiles (Ship, Launch Basket, Navigation Equipment Available, Warhead). Then the operator manually adds the match to the matching table. At the bottom left are warning tables that display the targets that cannot be reached (no missile can fulfill the hard constraints requirements), and the unused missiles. At the bottom right is a graphical summary of the current assignment, based on the matches included in the matching table. The horizontal bars fill in according to the number of targets assigned so far, with a breakdown by Target Priority. The operator can leverage a computer planning tool, Automatch, in which an algorithm instantly generates a mission-missile assignment and stores it in the matching table. Then, the operator has the option to manually modify this solution if deemed necessary. The heuristic search algorithm implemented in automatch sorts the missiles by priority. The missiles that have the fewest number of missions they can fulfill based on hard constraints are ranked first (this is to increase the number of assigned missions). Then, for each missile, the potential missions are prioritized in this order of importance: 1) loiter missions (the missile hovers over an area waiting for an emergent target to pop up), 2) high priority target, 3) medium priority target, and 4) low priority target. Firing rate and days to port information are not yet embedded in this search algorithm, but will in future developments of the software.

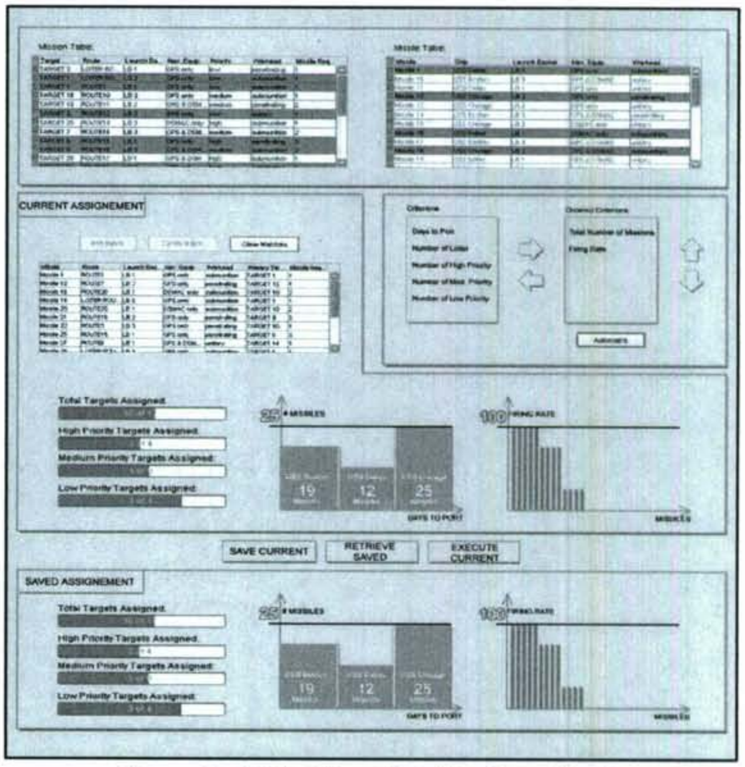


**Figure 2 - StrikeView Matching Interface 2.**

Interface 1 does not allow for any real collaboration between the human and the computer, only basic filtering. To provide a collaborative decision space, Interface 2 (Figure 2) allows operators the ability to leverage the computer's computational power, under human guidance. Interface 2 still includes the mission, missile, and matching tables, allowing for manual matching, and automatch is also available.

Whereas in Interface 1 the matching algorithm was completely hidden from the operator, in Interface 2 the operator can actually choose what criteria to include in the automatch, as well as a prioritization order between these criteria. Also, tick boxes next to the mission and missile tables enable the user to select a subset of missions and / or missiles to be considered by automatch. Furthermore, the assignment summary evolved to include, in addition to the horizontal bars, two other graphics that synthesize the assignment through the probabilistic (e.g. Firing Rate) and optimization (e.g. Days To Port) data. Finally, this interface includes a "save" option. When used, the current assignment is stored at the bottom of the screen, and a new assignment can be generated without modifying the saved assignment. This provides the user with a what –if comparison between two solutions.

Interfaces 1 and 2 are both based on the use of raw data. Interface 3 (Figure 3) is completely graphical and the user has no access to the mission and missile tables. The automatch button at the top is similar to that in Interface 1. However, the user can act on the level of prioritization of the probabilistic information (Firing Rate) and optimization information (Days To Port), in the automated algorithm, *via* the central screen sliding bar (the "prioritization bar") that represents what criteria (Firing Rate or Days To Port) should take precedence on the other in automatch.

The result of the assignment computed by automatch is displayed in two ways. First, the breakdown by mission priority (loiter, high, medium, low) in the four corners shows numerically and visually (position of the cursor in the vertical column) how many missions have been assigned, with a secondary breakdown by Warhead type. Then, the green area above and below the prioritization bar metaphorically represents the level of assignment: the more missions have been assigned, the more filled in the central area is. A complete assignment (all missions assigned) would be represented by a completely shaded central area. When the automatch solution is modified by the user, the new solution appears in green, and the first automatch appears as a pale gray in background, for comparison purposes.

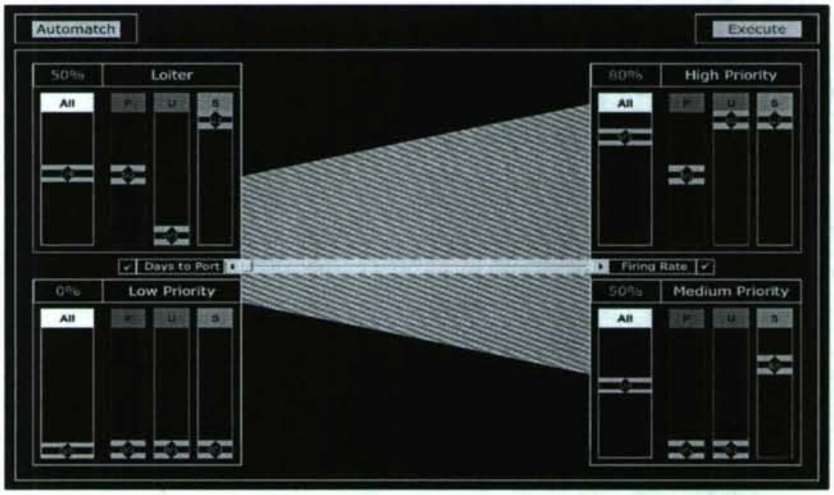


**Figure 3 - StrikeView Matching Interface 3.**

Additionally, the user can require the computer to search the solution space to accommodate specific needs: by clicking on the up or down arrows of the cursors in the vertical sliders, the user instructs the computer to find a way to increase or decrease the number of assignments corresponding to the specific slider. Automatch will then compute a new solution to accommodate for this requirement, by potentially modifying other assignments at higher priority levels.

In the experiment, six subjects participated in a cognitive walkthrough of the mission planning interfaces, including a former TLAM Strike Coordinator, an Air Force ROTC Cadet, an Army Infantryman with 18 years of experience, as well as three graduate students with extensive backgrounds in UAV operation and Human-Computer Interaction, two of them being USAF 2nd Lieutenants. A cognitive walkthrough evaluates how well a skilled user can perform novel or occasionally performed tasks. In this usability inspection method, ease of learning, ease of use, memorability, effectiveness and utility, among others, are investigated through exploration of the system.

Seven usability questions were used to rate the three interfaces on a Likert scale from 1 to 10:

1) How much perceptual effort is required to understand and use the interface?
2) How much mental processing is required to understand and use the interface?
3) How well would an operator perform with this interface?
4) How confused would an operator be using this interface?
5) How well does the interface give feedback to the user?
6) How much in control is the operator using the interface?
7) How satisfied vs. frustrated an operator would feel using the interface?

Two-tailed paired t-tests were performed on the ratings of the interfaces, between interfaces 1 and 2, 1 and 3, and 2 and 3. Using the Bonferroni criterion, the 0.05 level of significance was divided by three and results were therefore considered significant at the 0.016 level. We assumed that the parent population of the sample is normally distributed. Results are compiled in Figure 4 (significant differences between interfaces) and Figure 5 (no significant differences).

1) *Perceptual Activity* (Figure 4). The purely graphical interface (Interface 3) was considered to require less perceptual effort than Interfaces 1 ($p < 0.0004$) and 2 ($p < 0.003$). This result makes sense since the motivation behind the use of graphics is to minimize the need for and time spent on searching for information. But such an advantage has a cost. First, less information is available through the graphical interface, and then, the information is less precise, in that fewer parameters are visualized and accessible. Therefore, and as mentioned by the subjects, such a display would mainly be used for a rapid overview of the situation, with a few, simple interaction possibilities. This interface is good for conveying information, but insufficient for a comprehensive assignment task.

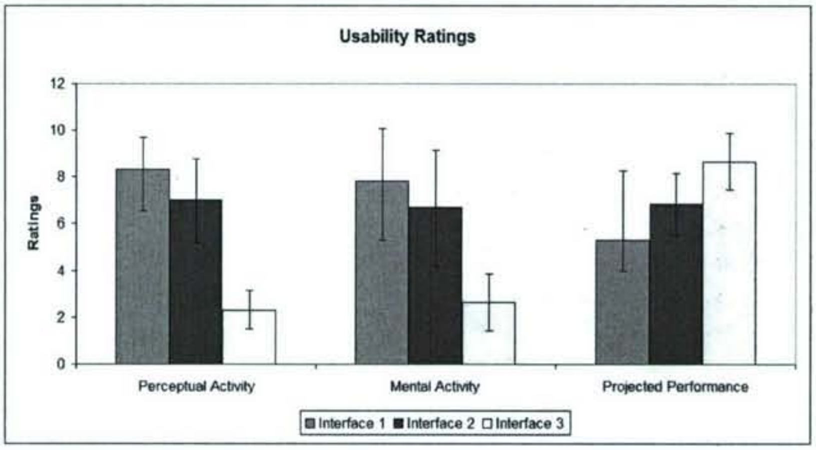


**Figure 4 – Significant Usability Ratings**

2) *Mental Activity* (Figure 4). Interface 3 required significantly less mental activity, such as thinking, deciding, calculating, remembering than Interface 1 ($p < 0.006$), and the difference with Interface 2 was almost significant ($p < 0.027$). This reinforces the perceptual activity results: a graphical interface is an efficient way to simply assess the situation without requiring the operator to add a mental process to build another layer of understanding. Indeed, using Interfaces 1 and 2 forces the user to interpret the data on the display: this delays the decision and is also subject to human errors, especially in a time-sensitive environment. In addition to ease of use and attractive to the eye, a graphical interface also simplifies the chain of cognitive processes required to understand and assess correctly the situation.

3) *Projected Performance* (Figure 4). The subjects estimated that Interface 3 would lead to better projected performance than Interfaces 1 ($p < 0.009$) and 2 ($p < 0.002$). But most subjects commented that the projected performance would be better with Interface 3 only if the instructions for the assignment were kept simple. With straightforward instructions, assignment

tasks would be done quickly and efficiently. However, as soon as the requirements and constraints for the task increase, the limitation of this interface would surface as detailed information and low level parameters are not accessible.

4) *Confusion* (Figure 5). No significant difference was found between the interfaces regarding the confusion they may generate. Interfaces were rated between a score of 1 (very confusing) and 10 (not at all confusing), and an increasing trend was found: although more visually simple than the table-based interfaces, the graphical interface tended to create more confusion. This may be the result of the inability of Interface 3 to control low level parameters. It is simple and efficient to use in a certain domain, but users' actions are limited: they may get confused because they do not know how to use the interface for specific action (or they do not know that they cannot do these actions). Raw data tables are less confusing because all information is available, and although the interface is more complex, once learned, it may not be as confusing.

5) *Feedback to the user* (Figure 5). This criterion was rated between 1 (poor feedback) and 10 (excellent feedback). A trend emerges from the results: the graphical interface seemed to provide better feedback to the user than Interface 2, which in turn was better than Interface 1. The system's response to user's action is key in the assessment of an interface: the operator needs to know that the intended performed actions have actually been performed. The graphical Interface 3 favors this criterion because change in the appearance of the screen as a result of the action is noticed more by the user than a change in the information inside a huge table of resources. Also, since Interface 2 provides more tools than Interface 1, and thus more feedback, it is understandable that its ratings are slightly higher.

6) *Control* (Figure 5). As expected, Interface 2 was considered the interface users were most in control of, mostly because more options are included in this interface. It is interesting to see that this control issue applies to "how many" actions the user can perform, and not "how much" the user can decide on the assignment. Indeed, it can be that the operator is provided with several automated tools, and hence feels "in control", while the real control is held by the computer in the way those tools are implemented (which is transparent to the user).

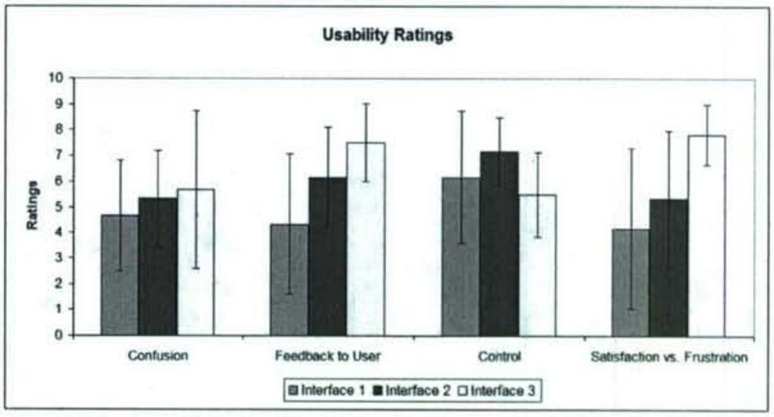


**Figure 5 – Non-significant Usability Ratings**

7) *Satisfaction vs. Frustration* (Figure 5). The rating scale went from very frustrated (1) to very satisfied (10), and an increasing trend amongst interfaces can be seen. Satisfaction progressively overcame frustration from Interface 1 to Interface 2 to Interface 3. This may be explained by the trends noticed in all other areas: with a graphical interface, the operator needs less perceptual and mental effort and is more in control, which contributes to an increased level of satisfaction. Conversely, with Interface 1, the range of possible actions was strongly restricted, hence causing frustration because of the inability for the users to do what they wanted.

### 3.2 Experiment #2[2]

As a result of the findings in experiment #1, the interfaces were modified to improve their usability and a formal experiment was conducted to determine the impact of proposed levels of human-computer collaboration on performance and cognitive strategies. To this end, twenty subjects from the Surface Warfare Officers School Command (SWOSCOM), at the Naval Station Newport, in Newport, RI, and from the Submarine Base New London in Groton, CT, participated in a formal experiment to test the three interfaces. These subjects (18 males, 2 females) were aged 25 to 37 (mean: 30 ± 2.6 sd) and had between 4 and 18 years of service in the U.S. Navy (mean: 8 ± 3.5 sd). While all had the same basic Navy strike training, two had extensive experience with TLAM Strike planning (more than 500 hours each), and seven had about 100 hours of experience each with TLAM Strike planning. Thirteen subjects had participated in live operations or exercises involving the use of Tomahawks, and three additional subjects had completed TLAM classroom training.

Five configurations of the StrikeView interfaces were tested: Interface 1 (I1), Interface 2 (I2), Interface 3 (I3), Interfaces 1 and 3 together (I13), and Interfaces 2 and 3 together (I23). Subjects were randomly assigned one interface configuration. Two scenarios involving the matching of 30 missions with 45 missiles were created which included a complete scenario (Scenario C), where at least one solution existed for the matching of all missions, as well as an incomplete scenario (Scenario I), where not all missions could be matched at the same time. Performance was evaluated using a weighted objective function of the number of matches accomplished by the operator, with a breakdown by priority.

Under Scenario C (all missiles have a matching mission), all twenty subjects reached the optimal performance score of 100 regardless of the interface configuration they used, which means that all subjects matched all missions, at all levels of priority. Under Scenario I however results were significantly different. Figure 6 plots the mean performance scores across subjects, categorized by interface configuration.

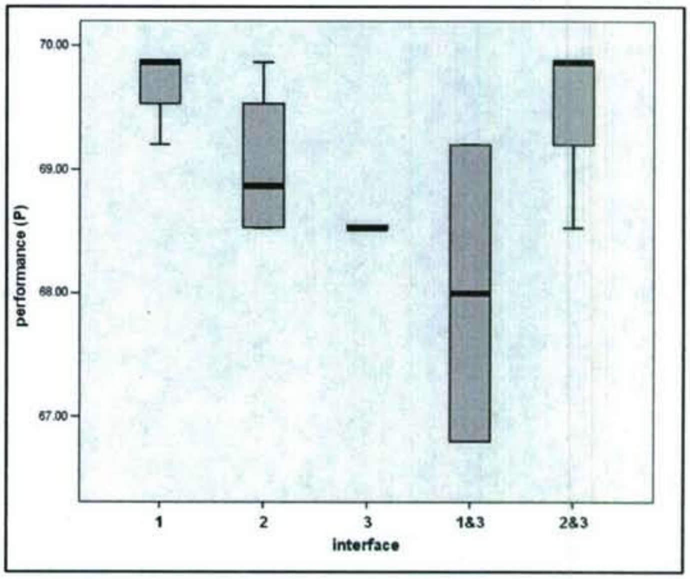


**Figure 6 - Performance results by interface, under Scenario I.**

For Scenario I, subjects using interface 1 (manual matching) or interfaces 2 and 3 together (collaborative matching and automatch) scored the best with an

[2] Detailed in Bruni, S., Cummings, M.L., "Tracking Resource Allocation Cognitive Strategies for Strike Planning" COGIS 2006 - Cognitive Systems with Interactive Sensors, Paris France.

average of 69.75, while those using interfaces 1 and 3 together scored the worst, at an average of 68.00. For interfaces used in combination, interfaces 2 and 3 together led to significantly better performance (average of 69.75) than interfaces 1 and 3 together (average of 68.00), which statistically significantly scored lower than any other interface configurations. However, while interface 3 performed the worst in terms of the single interfaces, Figure 6 shows that for Scenario I, subjects using interface 3 all reached the exact same performance: there was little deviation in performance for this condition.

In order to determine what cognitive strategies were implemented in solving this multivariate resource allocation problem and correlate this with performance, we developed the "Tracking Resource Allocation Cognitive Strategies" tool (TRACS) as a two-dimensional representation. The two axes, MODE and Level of Information Detail (or LOID) respectively correspond to the general functionalities as well as the information types available. For this mission planning software, the MODE axis includes the following functionalities: "browse", "search", "select", "filter", "evaluate", "backtrack", "automatch" and "update". The LOID axis is partitioned to correspond to the data used by the operator (above the x axis), while the lower y axis represents the criteria used to search the domain space. Within each sub-axis, LOID elements were ordered to reflect the level of abstraction of the information: "data item", "data cluster" (a group of data items with at least one common characteristic), "individual match" (a pair of matching data items, according to the search criteria), and "group of matches" (a cluster of individual matches). The criteria sub-axis featured, in order: "individual criterion" and "group of criteria".

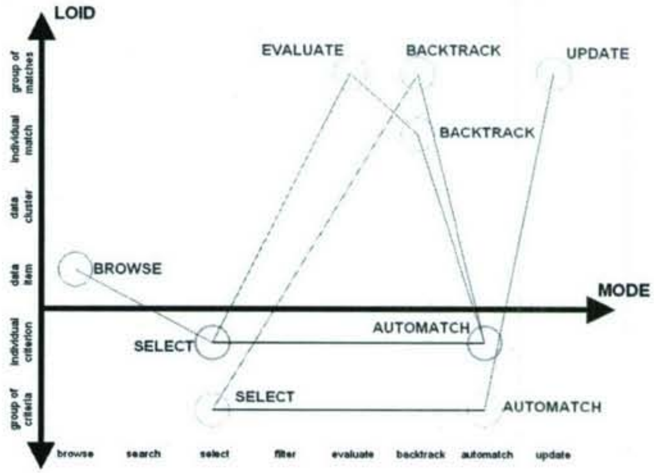


**Figure 7 - Example of a TRACS visualization**

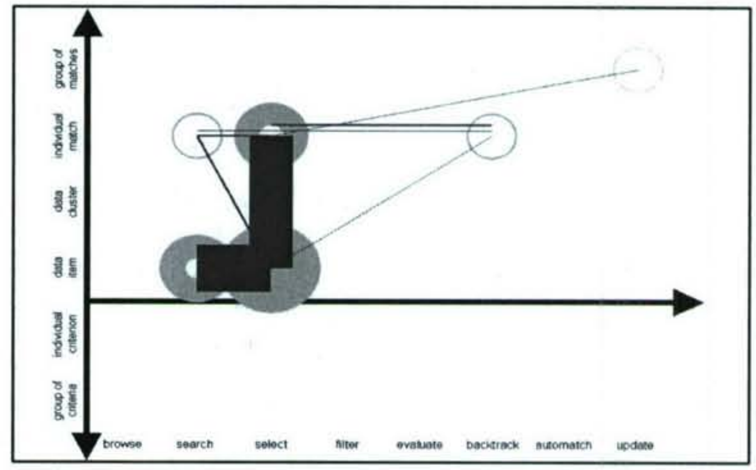


**Figure 8 - TRACS visualization, Interface 1, good performance**

Figure 7 displays an example of a cognitive strategy captured by TRACS. The underlying assumption while using TRACS is that every mouse click on the interface is considered as a conscious decision of the operator to interact with the DSS. Using a correspondence matrix for the two axes, we map each click (its location on the interface) to a specific MODE and LOID entry in the matrix. For each click, a circle is added to the corresponding cell in the TRACS visualization. If this cognitive step is a repeated action, the width of the circle is increased. Cognitive steps are connected to each other by a line when visited in sequence. Similarly, the thickness of these lines increases each time such a connection is repeated.

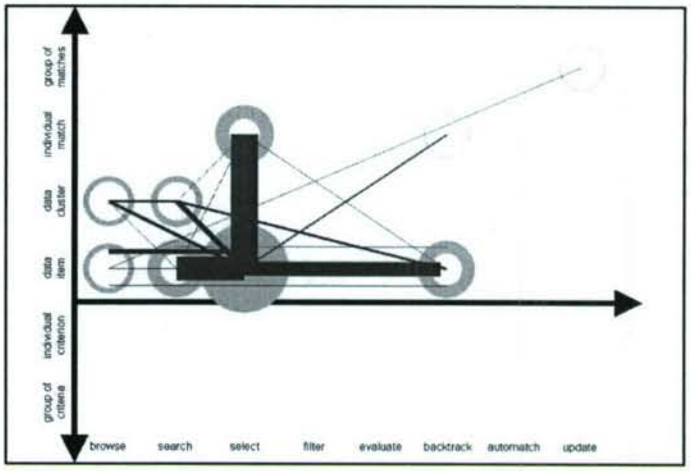


**Figure 9 - TRACS visualization, Interface 1, poor performance**

Figures 8 and 9 display the TRACS visualizations for two subjects who solved the incomplete scenario using interface 1. In terms of performance, the subject of Figure 8 outperformed that of Figure 9, and most significantly, the subject of Figure 8 validated the solution in 6 minutes while that of Figure 9 took over 26 minutes. In both TRACS representations, a similar pattern of cognitive steps emerges, linking the selection of individual matches, the selection of data items and the search for data items. Although this structure clearly constitutes the core cognitive strategy of in Figure 8, this same structure was weakly exhibited by the subject in Figure 9, who used several additional steps which led to poorer performance.

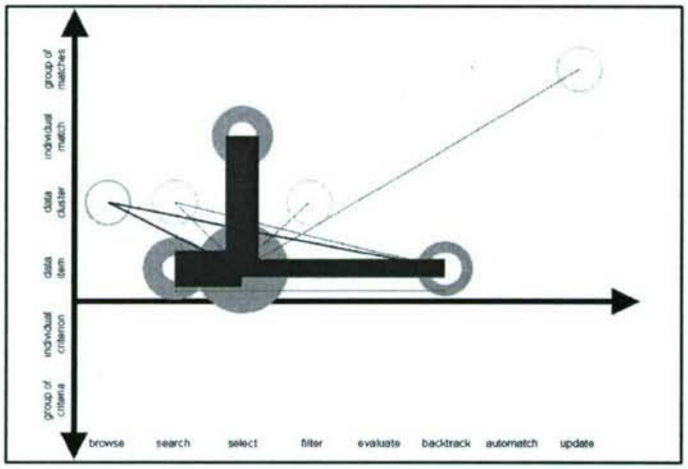


**Figure 10 - TRACS visualization, Interfaces 1 and 3, good performance**

Very similar TRACS visualizations were obtained when examining the cognitive strategies of subjects using the interface configuration featuring both interface 1 and 3. Figures 10 and 11 display the TRACS representations for these subjects, who respectively performed well and poorly on Scenario I. The subject of Figure 10 reached the best solution in less than 5 minutes whereas that of Figure 11, although coming within 5% of the best solution, took more than 20 minutes to complete the task. The subject of Figure 11 used the core strategy see previously, but to a lesser degree and secondary cognitive steps, such as browsing (of data items, data clusters and group of matches), or filtering (of data cluster), were repeatedly performed. Other additional cognitive steps can be seen in Figure 11 such as backtracking on group of matches (typically corresponding to the cancellation of the entire current solution), and automatch (the use of the heuristic search algorithm). The TRACS visualization for the subject in Figure 11 represents a very inefficient strategy, particularly in terms of time as compared to the subject in Figure 10.

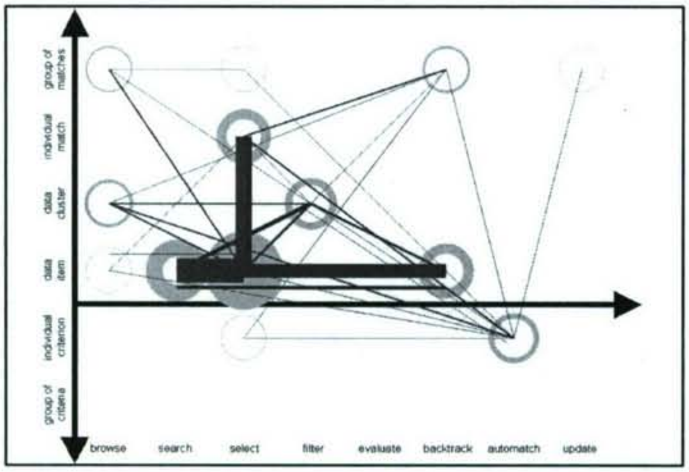


**Figure 11 - TRACS visualization, Interfaces 1 and 3, poor performance**

Figure 12 shows the TRACS visualization of a subject who solved the problem using interface 2 (collaborative matching) and reached the best solution. As magnified by the red square and triangle, this subject clearly exhibited two distinct strategies while solving the problem. First, he tried to solve it using the manual tools, such as the mission and missile tables. After browsing and searching this raw data, this subject switched to the automated tools, selected different criteria in order to implement the automatch capability and then evaluated the computer-generated solutions. The first strategy (manual matching) lasted ~3s before the subject decided to save that solution and switch to the second strategy (automatch) which lasted only 2s

These results demonstrate that regardless of their configuration, all three interfaces led to very good results, with performance averages per interface between 68 and 70 out of a possible 80. The fact that the performances were very close despite the levels of automation may however be a sign that the task on hand was no difficult enough to require full automation support. Indeed, the best performances on the incomplete scenario were obtained using the mostly manual interface (interface 1) or the combination of the collaborative and automatch interfaces (interfaces 2 and 3). On the other hand, when the automatch interface (interface 3) was paired with the manual interface (interface 1), performance decreased but remained acceptable. This shows that adding an interface that led to the best results (interface 1) to that which led to the lowest results alone (interface 3) did not lead to a better result, but to the contrary, it decreased performance.

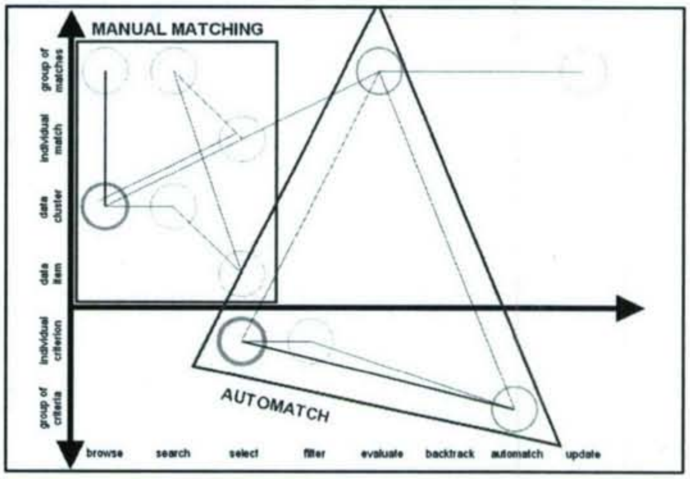


**Figure 12 - TRACS visualization, Interface 2, good performance**

### 3.3 Experiment #3

One of the specific focus areas of this research grant was to explore the difference in the way that humans interacted with automated planners in static versus dynamic conditions (i.e., under time pressure or not.) Once the first version of TRACS was completed, we recognized that modifications were needed to incorporate the time element. Figure 13 demonstrates that a temporal component was added in that a time bar and a playback feature were added across the bottom so that a researcher could replay what strategies occurred as a function of time (Bruni, Boussemart, & Cummings, 2007).

Once this revision was complete, we examined the effects of time pressure on the use of the automation during the strike planning process as described in the previous experiment. Our main research hypothesis was that people under temporal stress will rely more on automated tools in order to cope with the added workload. Time pressure is very relevant to the context of Command and Control (C2) since theaters of operations are inherently dynamic; the conjunction of changes and fixed deadline tend to put the operators under considerable stress due to the time-critical nature and the importance of the decision they have to make.

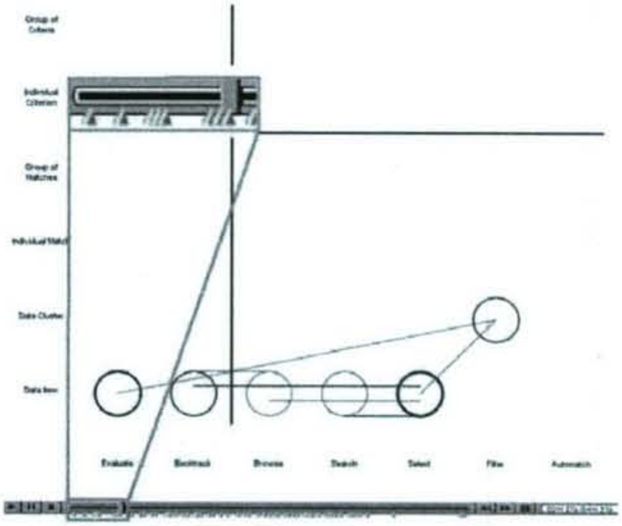


**Figure 13: TRACS with Temporal Features**

Using Interface #2 (Figure 2), an experiment was conducted using the same incomplete planning scenarios as described in the previous experiment, and subjects experienced two testing sessions: Distant Deadline (DD), a low time-pressure, baseline task with a 5 min limit, and Imminent Deadline (ID), which started just like the DD scenario with the same 5mn deadline, but at 3:30 mins, the subjects receive new orders to invert the priorities of the missions (low priority missions should be regarded as high priority and vice versa). This required the subject to re-plan the strike in the remaining 1mn30s, which corresponds to increased time pressure.

Sixteen subjects were tested, most of whom came from the MIT student population. When the number of calls to the automation was correlated with a performance score that measured submitted solution optimality, there were two significant Spearman-Rho correlations of note:

1) The number of calls to the automatch in the ID scenario and the final score in the ID scenario ($\rho$=0.740, p=0.001), which means that people who used automation in that scenario tended to do better.

2) The number of automatch calls between the ID and DD scenarios ($\rho$=0.855, p<0.0001), which suggests that some people are comfortable with using the automation and will tend to use it more often, whereas others will simply not use it. This confirms subjective evidence gathered during the post-experimental debrief.

To specifically examine the effect of time pressure, we divided the DD scenario between the first 3mn30s and the last 1mn30s in order to make direct comparisons with the ID scenario. A comparison of the total number of automatch clicks between the ID and the DD scenarios was significant (Mann-Whitney U Z=-2.558, p=0.011). These results were replicated when the number of automatch calls between the 2 phases of the ID scenario were compared (Z=-2.077, p=0.038). Thus, the majority of these calls came after the 3:30 change, so when the time pressure increased, subjects tended to use automation more.

Using four broad categories of cognitive strategies based on the use of automation (fully manual, mostly manual, mostly automated and fully automated), a non-parametric Mann-Whitney U test, revealed a significant difference in strategy between the ID and the DD scenarios (Z=-2.33, p=0.02). Those subjects who experienced the higher time pressure generally used automation more than those who did not. However, in the debrief session, multiple users reported that they should have used the automation after the change of ROE, but were overwhelmed by the additional workload and the time stress.

One other issue that this research raised was the impact of trust, or lack thereof, on performance. Some subjects, all MIT students, refused to use the automation because they didn't like to use an algorithm they were not familiar with and could generate suboptimal solutions. In essence, subjects tried to optimize the solution manually, but had difficulty when the new orders came in at 3:30. However, one subject was a very experienced US Air Force officer who designed flight plans with the aid of a computer. His experience and training taught him to trust the automation, and, according to the subject, even though the solution wasn't perfect it was considered to be "good enough". He was able to leverage the automation to create a plan that was accepted as good enough. While this needs to be investigated more fully, it appears that background and experience could significantly influence trust and use of automation in time-pressured environments.

While this research is preliminary (data analysis is still underway with the intent of publishing these results), the results support our main research hypothesis, namely that, under time pressure, subjects tend to use more automation than in a baseline, low temporal stress situation. This experiment also highlighted the link between trust, experience, and performance needs to be investigated further as this may provide insight as to the best transition path from platform to network-centric warfare.

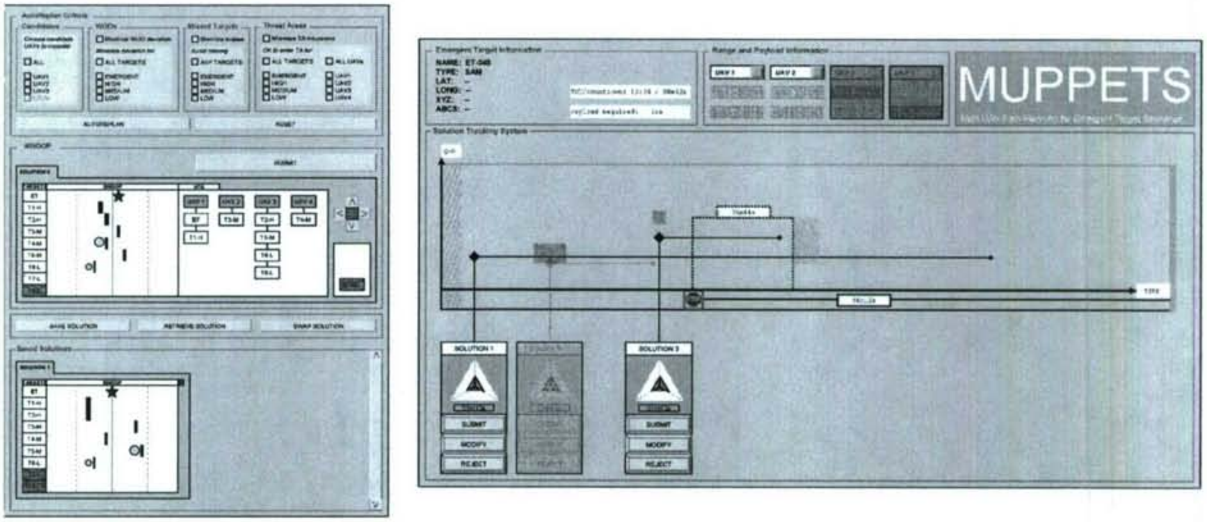
### 4.0 Time-Sensitive Targeting Interface Development

In order to more fully investigate the effects of time pressure on human-automation collaboration in a dynamic command and control setting, an initial prototype of a time sensitive targeting interface was created. This interface, based on the same mission planning environment as described earlier, allows operators to redirect either Tomahawk missiles or unmanned aerial vehicles (UAVs) to emergent (aka, pop-up) targets. The interface provides a geo-spatial map environment as well as the decision support in Figure 14. This decision support allows operators, at different levels of automation, to select one of many candidate UAVs or Tomahawks, while considering the effects on the overall mission in terms of reallocating the other vehicles to possible lost targets.

This is actually a very complex problem in that it is a moving horizon problem. The vehicles are moving very quickly and those solutions that exist at the current time may not exist even just minutes in the future. Moreover, reallocating a UAV from one set of targets to a higher priority emergent target will likely cause gaps in the overall air tasking order so this adds to the workload of the operator since they have to possibly replan all the other vehicle-target combinations in order to maximize overall mission success. So the time-sensitive targeting problem is a nested decision problem for the human – which vehicle should intercept the emergent target and how should the remaining vehicles be reallocated to maximize mission success?

While this grant ended at the same time the prototype was completed, this work has been extended through another ONR BAA: Human Supervisory Control Models for Command and Control of Unmanned Systems. Under this program, work is underway to embed two different artificial intelligence algorithms that represent increasing levels of automation (on par with those levels of automation represented in Experiments #1 & 2). Thus, as in these experiments, there are 3 interfaces: 1) Manual retargeting and re-routing, 2) Automation-assisted retargeting and re-routing, which relies on a human-guided heuristic search algorithm (Figure 14, left), and 3) a higher level automation generated solution using an anytime algorithm (Figure 14, right).

The decision support for the heuristic algorithm (Figure 14, left) provides windows of opportunity for not just the emergent target, but for all targets affected by the reallocation of missiles/UAVs. It also allows an operator to tailor a search for the best possible replan using multiple variables such as time on target, priority of targets, and minimization of threat exposure. This algorithm is not guaranteed to provide the best set of solutions (a common problem with heuristic algorithms), but it is very fast and allows the human operator to easily generate alternatives.



NAME: ET-048
TYPE: SAM
LAT: –
LONG: –
XYZ: –
ABCS: –

**Figure 14: Decision Support for Time Sensitive Targeting**

The decision support for the anytime algorithm (Figure 14, right) embeds a more complex algorithm that theoretically will provide the best possible solution, given enough time to solve the problem. This is an inherent problem with any algorithm that must solve a complex and large problem such as the multiple vehicle – multiple target case. The heuristic algorithm circumvents the time constraints but at the cost of solution quality. The anytime algorithm we will embed accounts not only for the best solution (i.e., maximizing targets engaged), but it also accounts for the cost of computation time. The decision support shows the operator how long the automation needs to plan to come up with the best possible solution, but it also shows the operator when other solutions of lesser quality could be available in advance of the most optimal solution. This algorithm and the human interaction with it is more complex and at a higher level of data aggregation than the heuristic algorithm so it remains to be seen how the less-than-optimal heuristic algorithm which may be easier to understand fares against the more-complex-but-more-accurate anytime algorithm in terms of human decision-making performance.

An experiment will be conducted this fall to determine how the different algorithms/automation levels impact human decision-making in the time-sensitive targeting environment (similar to that of Experiment #2 & #3). We will also use the TRACS tool previously described to investigate the associated cognitive strategies.

## 5.0 Collaborative Human-Computer Decision Making Model

As stated previously, one of the goals of this research effort was to develop a collaborative human-computer decision-making model that demonstrates how and what decision making functions should best be assigned to humans and computers in order to provide a mutually supportive human-computer decision making environment. To this end, we propose a framework that more accurately portrays collaborative decision-support systems beyond simply role allocation, termed the Human-Automation Collaboration Taxonomy or HACT. HACT provides a descriptive model to characterize and determine the degree to which a decision-support system is collaborative, for evaluation and comparison purposes.

We define collaboration as agents acting in a coordinated effort to solve a problem. An agent may be a human operator or an automated computer system, or "automation". HACT is only based on interactions between two agents (a human operator and automation). Typically, human-automation collaboration is an iterative process between the agents, and between the analysis and decision steps, which will be addressed in more detail in the next section.

While several taxonomies have been developed to classify and describe interactions between a human operator and a computer system, they are generally based on the concept of "level of automation". Despite some variations, these levels of automation, or LOAs, refer to the role allocation between the automation and the human (Parasuraman, Sheridan, & Wickens, 2000; Sheridan & Verplank, 1978; Wickens, Gordon, & Liu, 1998). These LOAs emphasize particular attributes, such as authority in the decision making process, solution generation abilities, or scope of action. The relative importance of each attribute can vary tremendously across command and control systems, hence, several scales have emerged, each typically focusing around one or two specific attributes.

There are certain elements of human-computer collaboration that are not addressed in any of the existing taxonomies. First, there is no mention of methods of whether or not the automation should be more transparent to the operator, in order to maintain mode awareness and detection of automation errors (Billings, 1997). Second, the exchange of information between agents is important in any form of collaboration. Many systems claim to be collaborative but the manner in which information is exchanged cannot be described as "mutual engagement," which is a key attribute for collaboration. Finally, systems where the level of automation could change with time either through human actions (adjustable autonomy) or independently (adaptive autonomy) are not considered (Goodrich et al., 2007). This unique characteristic of a potential decision support system should be considered as a step towards more elaborate forms of human-automation collaboration. HACT takes into account both the important attributes highlighted by previous LOAs and these missing attributes.

HACT features three steps: data acquisition, decision-making and action taking (Figure 15). The data acquisition step is similar to that proposed by Parasuraman et al. (2000): sensors get the information from the outside world or environment, and transform it into working data. First, the data from the previous step is analyzed, possibly in an iterative way where request for more data is sent to the sensors. The data analysis outputs some elements of a solution to the problem at hand. For example, in a mission planning situation, these elements of solutions may correspond to the current or projected status of some battlefield assets. The evaluation block will estimate the appropriateness of these elements of solutions for a potential solution. This block may initiate a recursive loop with the data analysis block. For instance, it may request more analysis of the domain space or part thereof to the data analysis block. At this level, sub-decisions are made to orient the search and analysis process.

Once the evaluation step is validated, i.e., sub-decisions are made, the results are assembled to constitute feasible solutions to the problem. In order to generate feasible solutions, it is possible to loop back to the previous evaluation phase, or even to the data analysis step. At some point, one or more feasible solutions are presented to a second evaluation step which will select one solution (or none) out of the pool of feasible solutions. After this selection procedure, a veto step is added, since it is possible for one or more of the collaborating agents to veto the solution selected (like in management-by-exception). If it is vetoed, the output of the veto step is empty, and the decision-making process starts over again. If the selected solution is not vetoed, it is considered the "final solution" and is transferred to the action mechanism for implementation.

Within the decision-making process of Figure 15, three key roles have been identified: Moderator, Generator, and Decider. In the context of collaborative human-computer decision making, these three roles are fulfilled either by the human operator, by automation, or by a combination of both. The Generator and the Decider roles involve parts of the model that are mutually exclusive: the domain of competency of the Generator (represented by the blue square to the left of Figure 15) does not overlap with that of the Decider (the green square to the right). However, the Moderator's role (represented by the red, dashed arrows in Figure 15) covers the whole decision-making process. Each role has its own scale, which lists the range of possible human-computer role allocations.

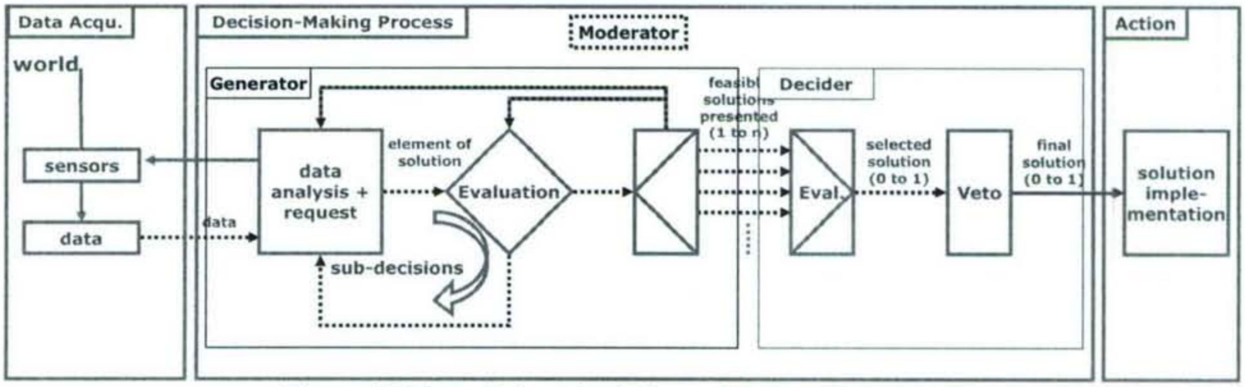


**Figure 15. The collaborative information-processing model**

## 5.1 The Moderator

The Moderator is the agent(s) that keeps the decision-making process moving forward (represented by the red, dashed arrows in Figure 15). The Moderator makes sure that the process goes from one block to another, and that the various phases are executed during collaboration. For instance, the Moderator may initiate the decision-making process and interaction between the human and automation. The Moderator may prompt or suggest that sub-decisions need to be made, or evaluations need to be considered. It could also be involved keeping the decision processing in pace when time deadlines must be met. The need for defining this role relates directly to ten-level Sheridan-Verplank LOA scale (1978), where the difference between LOA 4 and 5 is who initiates generation of a solution (Parasuraman et al., 2000). However, we recognize that this moderation occurs in multiple portions of the decision making process and separate from the task of generating solutions and selecting them.

## 5.2 The Generator

The Generator is the agent(s) that generates feasible solutions from the data. Typically, the Generator role involves searching, identifying, and creating solution(s) or parts thereof. Most of the previously discussed LOAs (Endsley & Kaber, 1999; Parasuraman et al., 2000) address the role of a solution generator. However, instead of focusing on the actual solution (e.g., automation generating one or many solutions), we expand the notion of Generator to include other aspects of solution generation, such as the automation analyzing data to make the solution generation easier for the human operator. Additionally, it is acknowledged that the role allocation for Generator may not be equally shared between the human operator and the automation. For example, the Generator could involve a system where the human defines multiple constraints and the automation searches for a set of possible solutions bounded by these constraints. However, a higher level Generator would be one where the automation proposes a set of possible solutions and then the human operator narrows down these solutions.

## 5.3 The Decider

The third essential role within HACT is the Decider. The Decider is the agent(s) that "makes the final decision", i.e. selects the potentially final solution out of the set of feasible solutions output by the Generator, and who has veto power over this selection decision. Veto power is a non-negotiable attribute: once an agent vetoes a decision, the other agent cannot supersede it. The veto power is also an important attribute that is described only in the Parasuraman et al. (2000) LOA scale (upper levels). These aspects are embedded in existing LOAs but they are mixed and incomplete.

The formulation of HACT essentially occurred at the conclusion of this research effort, with the results published recently at the 2007 International Command and Control conference in Newport, RI (Bruni, Marquez, Brzezinski, Nehme, & Boussemart, 2007). While the ONR grant has formally ended, work has continued on this model, now funded by AFOSR through an Architecture Science grant.

## 6.0 Mobile Advanced Command and Control System (MACCS) Status

With the 2006 award of a DURIP for a mobile experimental test bed, the Mobile Advanced Command and Control System (MACCS) was recently completed (Figure 16). While the award was announced in April, unfortunately contractual snags prevented any purchases to be made until July. Even with this delay, we have successfully purchased a 2006 extended Dodge Sprinter van, equipped it with GPS tracking alarm systems, and acquired the advanced displays and equipment. The van is now fully operational and we have held demonstrations at the Navy's ASNE Human Systems Integration Symposium in Annapolis, MD in March 2007, and also at NUWC and the ICCRTS conference in June at Newport, Rhode Island. MACCS was also recently featured in the ONR online newsletter, the Navigator (**http://www.onr.navy.mil/media/nre_navigator**). The first formal experiments are scheduled for the van in August and September in support of two ONR contracts, as well as an AFOSR contract.

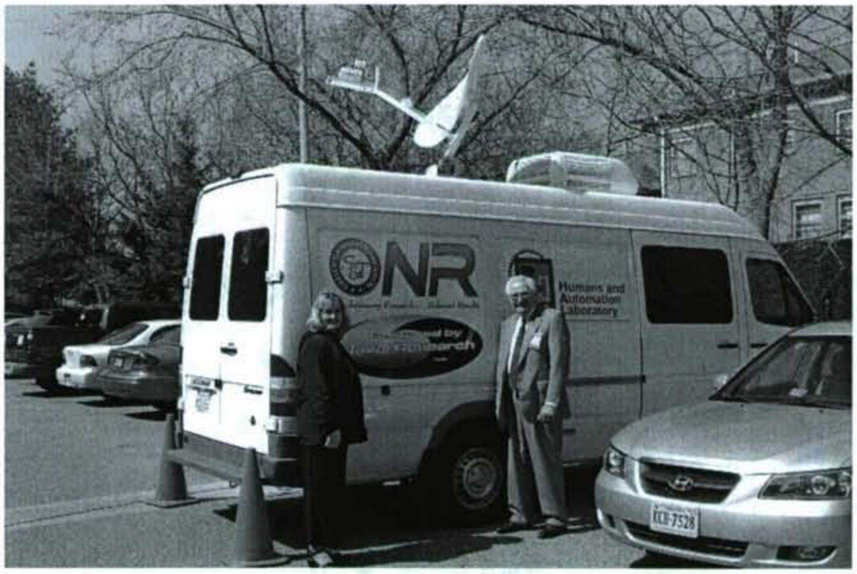


**Figure 16: MACCS on display at the 2007 ASNE Human Systems Integration Symposium**

## 7.0 Technology Transition Efforts

In an effort to broaden the impact of this research, significant work is underway to transition the lessons learned from this ONR project. These efforts include:

- Three ONR STTRs are underway that are directly leveraging the results from this project:
  - Plan Understanding for Mixed-initiative control of Autonomous systems (Partner: Charles River Analytics), in Phase II
  - Human-Directed Learning for Unmanned Air Vehicle Systems in Expeditionary Operations (Partner: Stottler Henke), in Phase I
  - Onboard Planning System for UAVs Supporting Expeditionary Reconnaissance and Surveillance (Partner: Aurora Flight Sciences), in Phase I
- Combat Systems of the Future Phase 2 SBIR with the Mikel, Assett Inc., and Rite Solutions (MARS) Coalition
- Capable Manpower Future Naval Warfighting Capability Human Systems Integration, ONR BAA 07-013, with the Mikel, Assett Inc., and Rite Solutions (MARS) Coalition (Contract will start in FY08).
- Human Supervisory Control Models for Command and Control of Unmanned Systems, ONR BAA (DEC06-NOV09)

- Architecture Science: Creating Visualizations for High Level Decision Makers, AFOSR BAA
- Collaborative Time Sensitive Targeting, Boeing Phantom Works
- Joint Warfighter Test and Training Capability Collaborative Metrics Applied to Manned Ground Vehicle Systems, US Army & Booz Allen Hamilton (contract in progress)
- A formal agreement for collaborative research and technology transition has been signed between the MIT Humans and Automation Laboratory and NUWC Newport.
- OCT 05 and DEC 06 briefings to the Navy's Strategic Studies Group in Newport, RI.

## 8.0 References

Billings, C. E. (1997). *Aviation Automation: The Search for a Human-Centered Approach.* Mahwah, NJ: Lawrence Erlbaum Associates.

Bruni, S., Boussemart, Y., & Cummings, M. L. (2007). *Visualizing Cognitive Strategies in Time-Critical Mission Replanning.* Paper presented at the ASNE Human Systems Integration Symposium, Annapolis, MD.

Bruni, S., Marquez, J. J., Brzezinski, A., Nehme, C., & Boussemart, Y. (2007). *Introducing a Human-Automation Collaboration Taxonomy (HACT) in Command and Control Decision-Support Systems.* Paper presented at the 12th International Command and Control Research and Technology Symposium, Newport, RI.

Endsley, M. R., & Kaber, D. B. (1999). Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics 42*(3), 462-492.

Gigerenzer, G., & Todd, P. M. (1999). *Simple Heuristics That Make Us Smart.* New York: Oxford University Press.

Goodrich, M. A., Johansen, J., McLain, T. W., Anderson, J., Sun, J., & Crandall, J. W. (2007). Managing Autonomy in Robot Teams: Observations from Four Experiments, *Human Robot Interaction.* Washington D.C.

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A Model for Types and Levels of Human Interaction with Automation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 30*(3), 286-297.

Sheridan, T. B., & Verplank, W. L. (1978). *Human and Computer Control of Undersea Teleoperators* (Man-Machine Systems Laboratory Report). Cambridge: MIT.

Simon, H. A., Hogarth, R., Piott, C. R., Raiffa, H., Schelling, K. A., Thaier, R., et al. (1986). *Decision Making and Problem Solving.* Paper presented at the Research Briefings 1986: Report of the Research Briefing Panel on Decision Making and Problem Solving, Washington D.C.

Tversky, A., & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and Biases. *Science, 185*(4157), 1124-1131.

Wickens, C. D., Gordon, S. E., & Liu, Y. (1998). *An Introduction to Human Factors Engineering*. New York: Longman.